On Optimization of Ad-blocking Lists for Mobile Devices

Saad Sajid Hashmi
saad.hashmi@hdr.mq.edu.au
Macquarie University
Australia

Muhammad Ikram
muhammad.ikram@mq.edu.au
Macquarie University
Australia

Stephen Smith
stephen.smith@mq.edu.au
Macquarie University
Australia

ABSTRACT

Online advertisements and third-party web tracking has gained much attention in recent years. Advertisers gather as much data and information about the users to provide targeted advertisement. Though this leads to a better user experience, it comes at the cost of privacy intrusive tracking. To this end, ad-blocking lists (or filter-lists, blacklists) have been introduced which prevent third-party tracking. Ad-blocking lists operate in a crowd-sourced manner, where new tracking domains (or rules) are continuously added by privacy activists and the redundant domains are discarded from the filter-list. Over time, the number of rules added outgrow the number of rules omitted, making it hard to manage the filter-lists. We empirically observe that the filter-lists mostly detect different ad and tracking domains. The filter-lists also use less than 1% of their rules on Alexa top 5,000 websites. This suggests the need to curate optimized filter-lists that provide high coverage and require less time to scan for a given domain on mobile devices. We develop an aggregated and filtered blacklist that is more than 150 times less bulky, and provides the same coverage as the union of the blacklists on Alexa top 5,000 websites. We also develop an update mechanism to incorporate new ad and tracking domains in the aggregated and filtered blacklist in a resource efficient manner.

CCS CONCEPTS

• **Security and privacy → Browser security**; **Web application security**; • **Human-centered computing → Empirical studies in ubiquitous and mobile computing**.

KEYWORDS

adblocking, coverage, efficiency, blacklists

ACM Reference Format:
Saad Sajid Hashmi, Muhammad Ikram, and Stephen Smith. 2019. On Optimization of Ad-blocking Lists for Mobile Devices. In *16th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services (MobiQuitous), November 12–14, 2019, Houston, TX, USA.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3360774.3360830

1 INTRODUCTION

Web tracking is the mechanism through which websites identify and collect information about users in the form of browsing history or other identifying information (like browser fingerprinting [31], ETags [28], and flash cookies [44]). Third parties (e.g., social widgets, advertisers, websites analytics engines) are embedded in the first party websites (that are directly visited by the users), to track users across different websites on the web. Third-party advertisements (in short ads) and web tracking activities have been carried out ever since the commercial world wide web has operated, to assist free services [40]. To this end, first-party websites leverage various techniques such as third-party `iframes` and JavaScript to show ads and track users' activities on websites. Although, third-party web tracking serves to bolster the business model of most content providers, its use in showing ads and building a list of websites the users have browsed, foments genuine privacy apprehensions. Additionally, many third-party services, such as ads, are being used as *malvertisements* [8, 37].

To remove intrusive ads and improve the web-page loading, users employ ad-blocking tools on desktop and mobile devices. The usage of ad-blocking tools on mobile devices has risen more rapidly in recent years, compared to the desktop browsers [20]. A number of studies have characterized and measured the ads and tracking ecosystem to protect user privacy and limit intrusive ads, resulting in numerous ad-blocking tools such as Ghostery [14], Adblock [2], Adblock Plus [3], Disconnect [6], and Privacy Badger [22] for the web [35] and mobile platforms [36]. Most of these tools are fueled by community-driven public blacklists (such as EasyPrivacy [11], EasyList [9], FanboyList [13], and hpHosts [15]) and are evaluated for their (in)effectiveness [34, 35, 49].

Though blacklists play an important part in protecting online users' privacy, their construction is largely ad hoc and unstructured. Most of the popular blacklists are maintained by ad-blocking tools developers and crowd-sourced users [47]. The blacklists comprise of rules and domains that need to be blocked or allowed (white-list). We demonstrate that most of the blacklists have very little in common with each other, and therefore, the domains blocked by the blacklists are also mostly different. As such, a single blacklist may not be able to provide protection against all the malicious ads. Several ad-blocking tools on the mobile platform employ multiple blacklists to provide better protection against ad and tracking domains [36]. But the composition of multiple blacklists comes at a cost of higher processing cycles, since the domain requested has to be compared against all the domains in the blacklist(s). An increase in processing time degrades the browsing experience of the user. To overcome this limitation, we develop a master blacklist by aggregating *useful* elements from the blacklists.

The main contribution of this paper is to develop an approach to aggregate, filter and curate blacklists to provide: *(i)* high coverage, and *(ii)* high efficiency (in terms of processing time). We conduct an empirical assessment of the coverage provided by the popular blacklists and demonstrate that aggregating blacklists by

MobiQuitous, November 12–14, 2019, Houston, TX, USA

ACM ISBN 978-1-4503-7283-1/19/11…$15.00
https://doi.org/10.1145/3360774.3360830

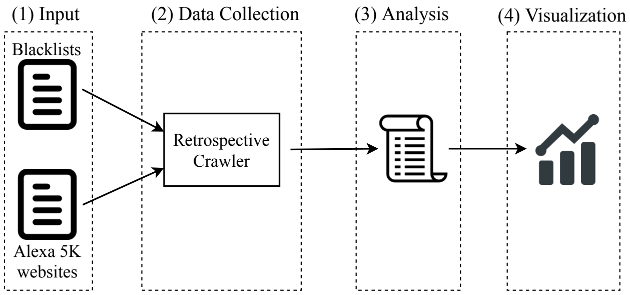


Figure 1: Overview of our approach.

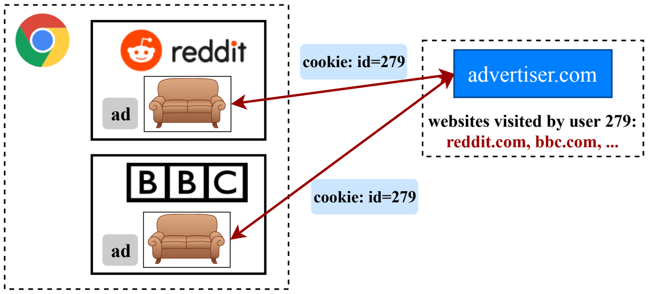


Figure 2: An example of cookie-based browser tracking [2]. The third-party tracking domain advertiser.com utilizes a cookie to track users on websites that embed advertisements from advertiser.com.

combining only the useful rules and domains significantly enhance the coverage while maintaining the efficiency to the level of lean blacklists. This is because we observe that most of the domains in a blacklist are not triggered in common browsing scenarios. We also develop a methodology for updating the aggregated blacklist by adding only the useful domains in it.

To conduct our empirical assessment, we leverage the historical data extracted from the Internet Archive's Wayback Machine [16] to analyze online ads and ad-blocking blacklists (listed in Table 1). From online ads, we look at the similarity of ad and tracking domains blocked by individual blacklists for the years 2015, 2016, and 2017. Figure 1 overviews our data gathering and analysis approach.

The rest of the paper is organized as follows: In Section 2, we briefly talk about web tracking and tracking prevention lists. Section 3 discusses the data collection and methodology of our analysis and Section 4 summarizes our analysis. In Section 5, we discuss related work and conclude our work in Section 6.

2 BACKGROUND

Online Tracking: The origins of online advertisement can be traced back to 1994, when a web magazine called HotWired advertised for American Telephone and Telegraph (AT&T) Corporation on its webpage [38]. Since then, the online ad revenue has gradually increased, both in absolute terms and as a percentage of all ad revenue [32]. A significant number of online businesses (i.e., Websites and free mobile apps) and their revenue hinge on online ads provided by third-parties. With recent technological advancements, third-party online ad and tracking services (websites' publishers and apps' developers, respectively) have embraced new ways to provide ads and re-identify users across domains. On the web platform, website publisher designates iframes to display ads and embed ad and tracking components (comprising of cookies and JavaScript codes from a third-party ad and tracking service) on their website [43]. The ad and tracking content of the iframe is separated from the hosting webpage, allowing only particular cross-frame interactions by the browsers [29]. The ad provider pays the website publisher based on the number of ad clicks or impressions, or both. Similarly, mobile developers leverage third-party tracking and advertisement libraries to track users activities and deliver advertisements [36]. These techniques re-identify users across domains for targeted ads [30]—advertising based on their purchase or search/browsing history—exposing users' sensitive details.

Personal Identifiable Information (PII) is stored by ad and tracking services as browser cookies, flash cookies, and HTML5 local storage and shared implicitly (via HTTP referrer) or explicitly (via tracker-provided JavaScript programs) with third parties. *First-party tracking* is when the user is tracked only by the first party domain (website which the user visits directly). First-party tracking can be used to count the repeated visits of a user to the website for *analytics*. *Third-party tracking* is when the user is tracked by a different domain on the website. For example, if the user visits bbc.com, and the advertisement on bbc.com is fetched from advertiser.com, then the user's browser sends a request to advertiser.com without the user being aware of it. Later, if the user visits reddit.com and the advertisement on reddit.com is also fetched from advertiser.com, then advertiser.com can build a browsing profile of the user and provide targeted advertisements based on the websites visited. Our study is based on the JavaScript programs of pupular websites (according to Alexa [1] top 5K wesbites ranking) that embed third-party domains. Therefore, in this paper, the terms 'ad and tracking' and 'third-party ad and tracking' are used interchangeably to mean third-party ad and tracking domains.

Defenses Against Online Tracking: Many ad-blocking tools have been created to check the privacy threat produced by third party ad and tracking services. They include pop-up blockers, privacy preserving web proxies (like Proxomitron [27], Privoxy [23], and Pi-hole [21]) to filter and block advertisement traffic at network-layer, and ad-blocking browsers (e.g., Brave [24]) or browser extensions (e.g., Adblock Plus, Disconnect, and Ghostery) that prevent web resources from rendering in the browser [35, 36]. Ad-blocking tools on the mobile platform may utilize Virtual Private Networks (VPN) tunnels [3] (like F-Secure Freedome [12], Perfect Privacy [26], and DashVPN [7]) to block advertisement related and malicious traffic, either locally on the mobile device or on remote servers, from all the installed applications.

Ad-blocking browser extensions that depend on filter-lists (also referred to as blacklists, ad-blocking lists, or tracking prevention lists) have become the most commonly used. The ad-blocking lists

[1] https://www.alexa.com/topsites
[2] Adapted from [40].
[3] https://developer.android.com/reference/android/net/VpnService.html

comprise of sets of URLs and domains of ad and tracking services [4]. The blacklists consist of both URLs and regular expressions to determine whether the resource being fetched is black-listed or white-listed (exception). These blacklists are either crowd-sourced with feedback from web users or maintained by the developers of ad-blocking tools [47]. Table 1 lists some popular ad and tracking lists utilized by ad-blocking tools.

3 DATA-SET AND METHODOLOGY

In this section, we provide an overview of the data-set and our measurement methodology. We begin by presenting the data-set used for experiments, formalize how we obtained it, and discuss our measurement methodology.

3.1 Data-set

We aim to study the coverage provided by ad-blocking lists against ad and tracking services in popular websites worldwide. Similar to our previous study in [34], we choose two sets of popular websites: *(i)* Alexa top 5K global websites (5K); and *(ii)* Alexa top 5K country-wise websites in the following fourteen countries: Australia (AU), Brazil (BR), Canada (CA), China (CN), Germany (DE), Israel (IL), India (IN), Iran (IR), Russia (RU), Saudi Arabia (SA), Singapore (SG), Ukraine (UA), United Kingdom (UK), and United States (US). Where not specified, the 'Alexa top 5K websites' in this paper refers to the union of both these sets of websites.

To conduct our longitudinal study from 2015 to 2017, we use web data from the Internet Archive's Wayback Machine. Since 1996, the Wayback Machine has archived full websites, including JavaScript codes, style sheets, and any multimedia resources that it can identify statically from the site's content. We refer to a single capture of a webpage (resp. ad and tracking domains in an ad-blocking blacklist) as a *snapshot.* Wayback Machine mirrors past snapshots of these websites on its own servers. We scrape the HTML DOM (Document Object Model, where HTML elements are defined as objects) of Alexa top 5K global and country-wise websites. The purpose of scraping the HTML DOM is to extract all JavaScript codes (along with their sources) from the websites, which is then used for our analysis. Wayback Machine has a number of archived snapshots, with varying intervals, for each website. We utilize Memento API [18] to capture snapshots at intervals of three months to detect most—if not all—of the ads and tracking domains that appeared on a website for a given year, during the period of 2015 to 2017. Memento API provides the nearest time-stamp for the archived snapshot of a website (resp. an ad-blocking blacklist) from the date provided.

We empirically observed that 85% of the snapshots are within an interval of 6 months. From these snapshots, we then obtain third-party sources from the embedded JavaScript codes, and by comparing it with the second level domains in ad-blocking blacklists, we obtain the ad and tracking domains in the crawled websites. For example, if `http://ad.doubleclick.co.uk/dot.gif` appears in the script tag of `sportsbet.com`, then the second level domain (`doubleclick.co.uk`) is extracted from the URL, and compared against the domains in the ad-blocking blacklists.

Similarly, by using Internet Archive's Wayback Machine, we obtain the snapshots of ad-blocking blacklists. We then compare the tracking domains observed in the Alexa top 5K websites and the blacklists. We say a domain in the blacklist is triggered if that domain is observed on any website.

To longitudinally analyze ad-blocking blacklists, we use the following blacklists used by ad-blockers (cf. Table 1):

Table 1: Description of the analyzed ad-blocking lists.

Filter-list	Description
AdAway [1]	Mobile ad/tracker hosts
Cameleon [5]	Ad/tracker hosts
EasyList [9]	Ad/trackers/analytics
EasyList_China [10]	Chinese EasyList websites
EasyPrivacy [11]	Trackers/analytics hosts
FanboySocial [13][5]	Social buttons/widgets hosts
hpHosts [15]	Ad/tracker/malicious hosts
Mahakala [17]	Ad/tracker hosts
MVPs [19]	Ad/tracker/malicious hosts

3.2 Methodology

The objective of this work is to aggregate and filter blacklists to provide *(i)* high coverage against threats posed by ad and tracking domains and, *(ii)* high efficiency in terms of time taken to process a network request. Before we aggregate blacklists, it is imperative to understand the *useful* elements (i.e., domains, urls, and 'rule') of a blacklist, and the *similarity* and *exclusive* contribution of blacklists.

Useful Elements of Blacklists: Useful elements of a blacklist are those elements that are triggered in common browsing scenarios. In our study we define a useful element as a domain (or rule) of a blacklist that is triggered when browsing the Alexa top 5K websites of any geo-location from §3.1. We measure the percentage of useful rules in each of the blacklist.

Rationale: An effective and efficient blacklist comprises of mostly useful rules that quickly determine if a web resource is to be blocked or not. Similarly, maintaining a blacklist requires addition of useful rules while ineffective rules are discarded.

Similarity of Blacklists: We define the similarity between any two blacklists as the number of ad and tracking domains in one blacklist that are present in the other, normalized by the size of the first blacklist. We measure the similarity of blacklist A with blacklist B as:

$$Sim_{A,B} = \frac{|A \cap B|}{|A|} \tag{1}$$

where A and B are the set of distinct ad and tracking domains in blacklists A and B, respectively. Thus, $Sim_{A,B} = 1$ means that every domain in blacklist A appears in blacklist B, while $Sim_{A,B} = 0$ means that no domain in blacklist A appears in blacklist B. We separately compute the similarity between blacklists for both the domains that comprise the blacklists and the domains that are detected by the blacklists in the Alexa top 5K websites of the analyzed countries. We then also measure the similarity in ad and tracking domains observed in the top 5K websites of various countries to demonstrate

[4] These lists also include malware, phishing and other annoyances such as pop-up dialog boxes that need to be blocked.

[5] http://www.fanboy.co.nz/fanboy-tracking.txt

that typical internet users in some of those countries mostly come across different ad and tracking domains.

Rationale: Ad-blocking blacklists comprise of ad and tracking domains. While the browser is fetching a resource from a third-party, the domain of that resource is compared with the domains in the blacklist to determine whether that resource should be loaded on the web-page or blocked. Since blacklists are maintained by crowd-sourced feedback from users, if multiple blacklists have high similarity, it may be possible that these blacklists have a large number of maintainers in common.

Exclusive Contribution of Blacklists: We measure the exclusive contribution of a blacklist A as:

$$Exclusive_A = \frac{\mid A \setminus U_{A'} \mid}{\mid A \mid} \tag{2}$$

where A is the set of distinct ad and tracking domains in blacklist A, and $U_{A'}$ is the union of ad and tracking domains of all the analyzed blacklists barring blacklist A. Thus, $Exclusive_A = 0$ means that every domain in blacklist A appears in some other blacklists, while $Exclusive_A = 1$ means that no domain in blacklist A appears in any other blacklist.

Rationale: The exclusive contribution of a blacklist is the proportion of ad and tracking domains that are unique to that blacklist. Exclusive contribution informs us how much a blacklist is different from the rest.

Coverage of Blacklists: For our first objective, we compare the second-level domains of JavaScript sources (observed in retrospectively crawled top 5K websites from the years 2015 to 2017) with those of known ad and tracking domains from the ad-blocking blacklists. This comparison reveals the number of ad and tracking domains blocked in different ad-blocking blacklists' yearly snapshots. We aim to investigate the coverage provided by each blacklist. Since the tracking ecosystem is continuously evolving, there is no ground truth available on total coverage of ad and tracking domains. We therefore propose two metrics, *coverage_α* and *coverage_β*, to measure the coverage provided by each blacklist against ad and tracking domains. We define coverage_α as the ratio of the number of distinct domains detected by blacklist A over the number of distinct domains detected by all the blacklists. Coverage_ α of blacklist A is measured as:

$$Coverage\_\alpha_A = \frac{\mid A \mid}{\mid U \mid} \tag{3}$$

where A is the set of distinct ad and tracking domains detected by blacklist A and U is the set of distinct ad and tracking domains blocked by the union of all blacklists. Unfortunately, there is no systematic way of evaluating the exact accuracy of each domain in a blacklist to obtain the ground truth. It may be possible that a blacklist contains some false positives. To tackle this limitation, we propose coverage_β metric, which is measured as the ratio of the number of unique domains detected by a blacklist over the number of unique domains detected by all the blacklists, with a condition that the domain occurs in at least two blacklists. Let $A_1, A_2, A_3, \ldots$ represent all the blacklists. Then coverage_β of blacklist A_i is measured as:

$$Coverage\_\beta_{A_i} = \frac{\mid A_i \cap U \mid}{\mid U \mid} \tag{4}$$

where $U = \{x : \exists\ i, j, \quad i \neq j,\ \ x \in A_i \cap A_j\}$ and A_i, A_j are different blacklists.

Efficiency of Blacklists: Blacklists are maintained via crowdsourcing by privacy activists. Contributors adding more domains to the blacklists provide better coverage. The downside of this growth in coverage is that it requires more resources to manage and filter out false positives and rules that are not useful.

Vastel et al. [47] have shown that popular blacklists add new domains (and rules) 1.7 times more often than removing old domains (and rules). This suggests that the blacklist may be accumulating not so useful rules over time, as websites embed new ad and tracking domains over time either because the ad and tracking domain is not functioning anymore or to evade ad-blocking tools [50]. As a result, the increase in coverage provided by blacklists comes at higher maintenance costs, and thus may increase the time taken by blacklists to allow or block a network request. Ad-blocking tools decide whether to allow or block a network request by matching the URL of the request against every network rule. Thus blacklists comprising of a large number of network rules may degrade the browsing experience of users, particularly on mobile devices.

We measure the efficiency of a blacklist as the time taken (in seconds) to read all the domains in a blacklist. For measuring efficiency, we use the emulator of Google Pixel 3 (*see* [25] for technical specifications of this device) with API level 26 (Android 8.0) provided by Android Studio 3.4.1 [4]. For each blacklist, we read all the domains 1,000 times and measure the total time taken.

Aggregating and Filtering Blacklists: To provide better coverage with high efficiency, we propose an aggregated blacklist that combines the useful elements from the individual blacklists and filters out unproductive domains to improve the efficiency. The proposed aggregated [6] blacklist provides the same coverage as would have been provided by the union of all blacklists, but is far leaner than the union of blacklists.

The ad and tracking domains that occur only in the tail of the websites (Alexa rank > 5K) and are detected by the individual blacklists, will evade the proposed aggregated blacklist. To overcome this limitation, and also to detect new ad and tracking domains that emerge on the websites, we propose an offline approach for maintaining the aggregated blacklist. Figure 3 shows the overview of maintaining the aggregated blacklist. All the third-party requests a browser sends go to the aggregated ad-blocker. If the third-party domain is in the blacklist, the request will be blocked, else the request will be allowed. The allowed third-party domain will then be passed to on offline checker. The offline checker comprises of all the domains that are in the union of blacklists, and verifies the domains passed to it in the offline mode. This is done to ensure that the aggregated blacklist does not take a large amount of time in processing a network request and thus, the browsing experience of the user is not degraded. If the third-party domain requested by the browser is in the offline checker, then that domain is added to the aggregated blacklist, and will be blocked the next time that domain is requested by the browser.

To summarize, we start from an initial seed of useful domains in our aggregated blacklist. If the user is concerned about the false

[6] In this work, we use the terms 'aggregated blacklist' and 'aggregated and filtered blacklist', interchangeably.

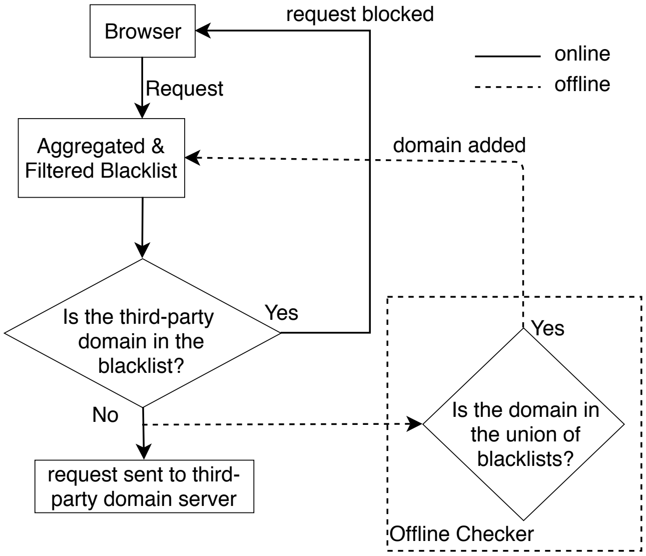

Browser
request blocked
online
offline
Request
Aggregated & Filtered Blacklist
domain added
Is the third-party domain in the blacklist?
Yes
No
request sent to third-party domain server
Yes
Is the domain in the union of blacklists?
Offline Checker

Figure 3: An overview of how the Aggregated and Filtered Blacklist updates itself. If a third-party domain requested by the browser is not in the blacklist, that domain is sent to the offline checker where it is compared against the domains in the union of the blacklists. If it is found there, then we add that domain in our aggregated blacklist and the domain request will be blocked in the next cycle.

positives, they can start with a smaller initial seed comprising of only those domains that occur in at least two blacklists (*see* Eq.4). The aggregated blacklist is then custom maintained by adding useful domains based on the browsing behavior of individual users. This way, only the useful elements of a blacklist are picked for the aggregated blacklist, and the domains that are never encountered by the user are not added.

To show consistency of our approach, we repeat the experiments three times on the data collected for the years 2015, 2016, and 2017.

4 ANALYSIS AND RESULTS

Using the dataset (cf. §3.1), in this section, we analyze the effectiveness of our approach.

Useful Elements of Blacklist: We determine the proportion of useful elements in the analyzed blacklists. Table 2 lists the annual ratio of useful elements in each of the blacklists. We observe that blacklists that consist of more than 10K domains have less than 1% useful domains. This shows that more than 99% of the rules that are added in these blacklists provide no benefit to the users in common browsing scenarios.

Similarity of Blacklists: We compute the similarity among the analyzed blacklists. Figures 4 and 5 show the similarity in the domains that comprise the blacklists, and the domains that are detected by the blacklists, respectively. We observe that the ratios of common domains (both that the blacklists are comprised of and those that they detect) is very low for most of the blacklists. This suggests that blacklists utilize different sources for curating themselves. Since the blacklists detect mostly different ad and tracking domains, it is possible to miss some tracking domains that are not listed in a particular blacklist. This highlights our need to aggregate blacklists to provide better coverage against ad and tracking domains. Note that the matrix is asymmetric, because, in general, $\frac{|A \cap B|}{|A|} \neq \frac{|A \cap B|}{|B|}$. For example, the shade of the second cell (from left) in top row of Figure 4(a) shows the value of $\frac{|AdAway \cap Cameleon|}{|AdAway|}$, whereas the shade of first cell in the second row of the same figure represents the value of $\frac{|AdAway \cap Cameleon|}{|Cameleon|}$.

Table 2: Ratio of useful domains in the ad-blocking lists in different years.

Blacklist	Year	# Domains Consisting of	# Domains Detected	% Detected (Useful)
AdAway	2015	132	34	25.76
AdAway	2016	123	29	23.58
AdAway	2017	124	27	21.77
Cameleon	2015	670	128	19.1
Cameleon	2016	640	126	19.69
Cameleon	2017	513	86	16.76
EasyList	2015	1206	188	15.59
EasyList	2016	2012	246	12.23
EasyList	2017	1628	79	4.85
EasyList_China	2015	2153	252	11.7
EasyList_China	2016	3702	298	8.05
EasyList_China	2017	1582	77	4.87
EasyPrivacy	2015	510	111	21.76
EasyPrivacy	2016	581	139	23.92
EasyPrivacy	2017	331	97	29.31
FanboySocial	2015	2826	352	12.46
FanboySocial	2016	2826	327	11.57
FanboySocial	2017	18	3	16.67
hpHosts	2015	116647	652	0.56
hpHosts	2016	79478	293	0.37
hpHosts	2017	220646	1052	0.48
Mahakala	2015	594667	2519	0.42
Mahakala	2016	1112266	1418	0.13
Mahakala	2017	139265	195	0.14
MVPs	2015	645	138	21.4
MVPs	2016	160	19	11.88
MVPs	2017	403885	1973	0.49

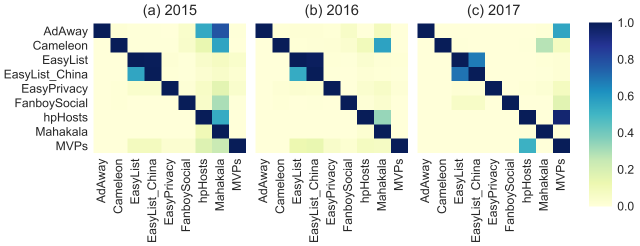


Figure 4: Similarity in the ad and tracking domains that comprise the blacklists.

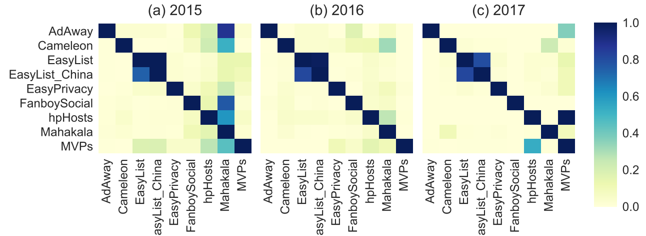


Figure 5: Similarity of ad and tracking domains that are detected by the blacklists.

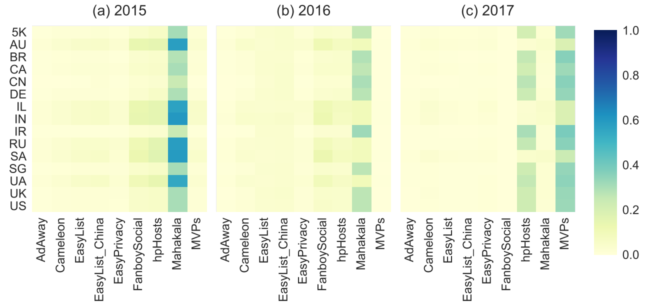


Figure 6: Similarity of ad and tracking domains observed in the Alexa top 5K websites of countries with the blacklists.

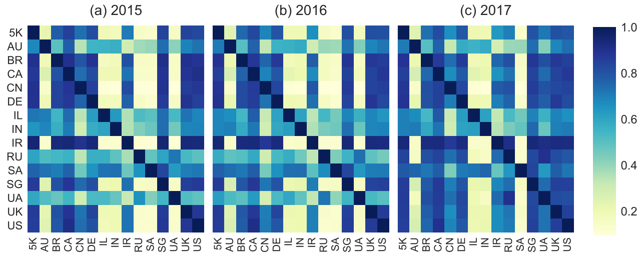


Figure 7: Similarity of ad and tracking domains in the top 5K websites of different countries.

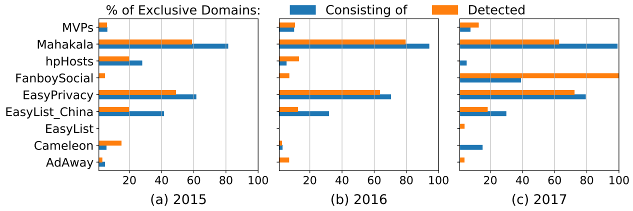


Figure 8: Exclusive contribution of the blacklists.

Figure 6 shows the similarity of ad and tracking domains observed in the Alexa top 5K websites of countries with the domains in the blacklists. We notice that the analyzed blacklists are detecting ad and tracking domains uniformly across countries. Interestingly, EasyList_China that is curated for Chinese websites, does not have the highest coverage for Alexa top 5K websites in China (CN). Also, the low similarity values for different countries suggest that a single blacklist can not provide high coverage, either in a single country or across countries, barring Mahakala in 2015. But Mahakala is also the bulkiest of all the blacklists and thus the least efficient in terms of time taken to process a request.

Figure 7 shows the similarity of ad and tracking domains observed in the Alexa top 5K websites of various countries for the years 2015-17. We notice varying degrees of similarity among countries. Russia (RU), Saudi Arabia (SA), and Ukraine (UA) have least similar domains with the other countries in 2015 and 2016, and Saudi Arabia having least similar domains in 2017.

Exclusive Contribution of Blacklists: Exclusive contribution represents the ratio of ad and tracking domains in a blacklist that are not present in other blacklists. It represents the uniqueness of a blacklist. Blacklists having a high exclusive contribution suggest that they utilize unique sources (crowd-sourced users, ad-blocking tools developers) for updating. Figure 8 shows the percentage of domains that are exclusive to each blacklist for the years 2015-17. We notice that Mahakala and EasyPrivacy have a high exclusive contribution (>48%) in both, the domains that these blacklists consist of as well as the domains they detected, for the given three years. In Figure 8(c), we observe that FanboySocial has a 100% exclusive contribution in 2017. This implies that the three ad and tracking domains that were blocked by FanboySocial in 2017 (*see* Table 2) were not detected by any other blacklist.

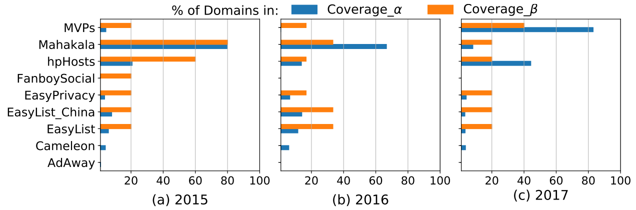
Coverage provided by the Blacklists: The coverage of a blacklist is the number of blocked ad and tracking domains by that blacklist, expressed as a proportion of all blocked domains from the union of blacklists. The coverage_α and coverage_β metrics provide a quantitative measure of how well a blacklist provides protection against the ad and tracking domains (*see* Eq.3 and 4). As mentioned previously, due to unavailability of ground truth, we take the union of the analyzed blacklists as the global coverage. There is also the possibility of false positives being added in a particular blacklist, thus preventing the browser from loading the functional components. To overcome this limitation, we define the global coverage as the union of all blacklists with the added constraint that each domain in the union exists in at least two blacklists (cf. Eq.4).

Figure 9 shows the coverage provided by the blacklists for the years 2015-17. Mahakala provides the highest coverage_α (and coverage_β) in 2015 and 2016, whereas MVPs provides the highest coverage_α (and coverage_β) in 2017. We notice that there is no fixed pattern of the observed values of coverage_α and coverage_β for the individual blacklists. The difference in coverage_α and coverage_β for the individual blacklists is because either the blacklist is blocking new ad and tracking domains that are yet to be detected by the other blacklists or these blacklists are blocking false positives. Moreover, not a single blacklist provides consistently high coverage_α or coverage_β for all the three years, thus highlighting the need to aggregate the blacklists.

Efficiency of Blacklists: We measure the efficiency of a blacklist in terms of time taken (in seconds) to decide whether the third-party domain requested by the browser is blacklisted or not. Less amount of time taken implies that the blacklist is efficient, whereas

Table 3: Efficiency (in seconds) and coverage_α of blacklists in different years. Blacklists that provide high coverage_α also consume more time in processing requests, thus making them inefficient for mobile devices.

Blacklist	2015		2016		2017	
	Efficiency (s)	Coverage_α (%)	Efficiency (s)	Coverage_α (%)	Efficiency (s)	Coverage_α (%)
AdAway	0.22	1.05	0.18	1.19	0.18	1.13
Cameleon	0.34	3.97	0.34	5.19	0.29	3.59
EasyList	0.45	5.83	0.68	10.12	0.54	3.3
EasyList_China	0.62	7.81	0.97	12.26	0.52	3.21
EasyPrivacy	0.25	3.44	0.28	5.72	0.23	4.05
FanboySocial	0.77	10.91	0.77	13.46	0.13	0.13
hpHosts	27.1	20.22	18.98	12.06	54.8	43.89
Mahakala	138.1	78.11	251.24	58.35	32.24	8.14
MVPs	0.29	4.28	0.17	0.78	96.6	82.31
Union of the blacklists	163.4	100	290.65	100	147.1	100
Aggregated & Filtered	0.94	100	0.97	100	0.89	100



(a) 2015

(b) 2016

(c) 2017

Figure 9: Coverage_α and Coverage_β provided by the individual blacklists. No blacklist consistently provides high coverage_α or coverage_β in the given period.

high amount of time means that the blacklist is inefficient for mobile devices. For our measurement study, we use the emulator of Google Pixel 3 with API level 26 (Android 8.0) that is provided by Android Studio 3.4.1. For the empirical measurement, we test for the worst case scenarios of each blacklist, and therefore read all the domains in the blacklist. We repeat reading all the domains 1,000 times and compute the total time taken in seconds. Table 3 lists the coverage and efficiency of the individual blacklists, the union of all blacklists, and the proposed aggregated and filtered blacklist for the period 2015-17.

The proposed aggregated and filtered blacklist provides the same coverage as the union of all blacklists to the users in common browsing scenarios, and is also more than 150 times efficient. The blacklists that have comparable efficiency levels with the aggregated blacklist provide far less coverage_α (<15%). For the first cycle, the aggregated and filtered blacklist does not block ad and tracking domains observed only in websites with Alexa rank greater than 5K. If any of those domains exists in the union of all blacklists, then the offline checker will add that domain to the aggregated blacklist. In the next cycle that domain will be blocked by the aggregated blacklist.

5 RELATED WORK

Online Tracking: Krishnamurthy and Wills [39], use longitudinal measurement snapshots to show that the third-party ad and tracking domains usage was already increasing significantly between 2005 and 2008. Their work provides a comprehensive overview of how some organizations (e.g., Google) have increased their tracking coverage both by increased usage of some of the third-party domains that they own, and through active acquisition of new domains (e.g., DoubleClick) that provide ads services, analytics, and tracking.

Roesner et al., [43] categorize mainstreaming tracking and analytics services and studied their prevalence in the wild. Ikram et al., [35] evaluate the usability of five different tracking prevention plugins and reveal that contemporary blacklists based tracking prevention tools are inefficient and have high false positives.

Curating Blacklists: Similar work in the field of cyber-attacks (like malware, spam) has been done by Ramanthan et al. [42]. They aggregate blacklists by estimating false positives over different blacklists and IP-address regions to achieve high recall with a minimal loss in specificity. A closer work is by Li et al. [41] where they propose metrics to measure the indicators of host or network that may be compromised. They did an empirical evaluation of several paid and open-source data feeds. Vastel et al. [47] analyzed EasyList and identified that it mostly accumulates useless (or dead) rules. They proposed optimization on EasyList to improve its performance on resource-constrained devices. Gugelmann et al. [33] proposed a classifier for detecting privacy intrusive web services using HTTP traffic statistics to support the manually curated ad-blocking blacklists.

Lognitudinal Measurement Studies: Several studies have used Wayback Machine for retrospective analysis. Soska et al., [45] used data mining and machine learning based techniques to anticipate if a website will become malicious over time. Wang et al. [48] performed a longitudinal study on search-engine optimization campaigns by infiltrating a search poisoning botnet. Vasek and Moore [46] examined the characteristics of compromised webservers. Their study revealed that websites using a content management system (CMS) are more prone to be compromised. They further deduced that popular versions of CMS are more likely to be targeted, and thus have a greater risk of compromise. Hashmi et al. [34] performed a retrospective analysis on various ad-blocking blacklists and showed that the number of ad and tracking domains on websites change over time. They also measured the effectiveness of ad-blocking lists in terms of rate of change and update speed of the blacklists.

6 CONCLUSION

In this paper, we present some of the limitations for current blacklists employed by the ad-blocking tools. We argue that the accumulation of stale rules in ad-blocking lists increases the processing time for scanning a web request, therefore, negatively affecting the browsing experience of the users. Current ad-blocking blacklists also fail to provide consistently high coverage. In the years where some blacklists provide high coverage, they do consume large amounts of processing time thus making them inefficient. To overcome this limitation, we develop an approach to aggregate and filter blacklists that provide high coverage with high efficiency. We aggregate the useful components of different blacklists and filter out the not so useful parts. In common browsing scenarios, the aggregated blacklist provides the coverage that is equivalent to the coverage provided by the union of all blacklists but is more than 150 times efficient. Some of the individual blacklists consume nearly as much time as the aggregated blacklist but provide far less coverage (<15%) compared to the aggregated blacklist. In scenarios where our aggregated blacklist encounters a new tracking domain, we develop an offline approach to verify the correctness of this domain against the union of all the blacklists. This ensures that only the useful elements are added in the aggregated blacklist and the processing time remains minimized, therefore not degrading the users' browsing experience on mobile devices.

7 ACKNOWLEDGEMENT

This research was supported by the Macquarie University Research Training Program (RTP) Scholarship.

REFERENCES

[1] 2019. AdAway Hosts. https://adaway.org/hosts.txt.

[2] 2019. Adblock: content filtering and ad blocking browser extension. https://getadblock.com.

[3] 2019. Adblock Plus: open-source browser extension for content-filtering and ad blocking. https://adblockplus.org.

[4] 2019. Android Studio - tools for building applications on every type of Android device. https://developer.android.com/studio.

[5] 2019. Cameleon. http://sysctl.org/cameleon.

[6] 2019. Canonical repository for the Disconnect services file. https://disconnect.me/trackerprotection.

[7] 2019. Dash VPN homepage. http://dashvpn.io.

[8] 2019. Dasient Smart Web Security Q3 2010 Malware Update. http://blog.dasient.com/2010/11/normal.html.

[9] 2019. EasyList. https://easylist.to/easylist/easylist.txt.

[10] 2019. EasyList China+EasyList. https://easylist-downloads.adblockplus.org/easylistchina+easylist.txt.

[11] 2019. EasyPrivacy. https://easylist.to/easylist/easyprivacy.txt.

[12] 2019. F-Secure Freedome VPN homepage. https://www.f-secure.com/en/home/products/freedome.

[13] 2019. Fanboy's Social Blocking List. https://easylist-downloads.adblockplus.org/fanboy-social.txt.

[14] 2019. Ghostery: open-source browser extension and mobile browser application to control JavaScript tags. https://www.ghostery.com.

[15] 2019. hpHosts Online – Simple, Searchable & Free! http://www.hosts-file.net.

[16] 2019. Internet Archive: Digital Library of Free & Borrowable Books, Movies, Music & Wayback Machine. http://archive.org.

[17] 2019. Mahakala Adblocking Hosts. https://adblock.mahakala.is.

[18] 2019. Momento: Time Travel. http://timetravel.mementoweb.org.

[19] 2019. MVPs Hosts Lists. http://winhelp2002.mvps.org/hosts.txt.

[20] 2019. PageFair's 2017 Global Adblock Report. https://pagefair.com/downloads/2017/01/PageFair-2017-Adblock-Report.pdf.

[21] 2019. Pi-hole: A black hole for Internet advertisements. https://pi-hole.net.

[22] 2019. Privacy Badger: open-source browser extension for blocking advertisements and tracking cookies. https://www.eff.org/fr/node/99095.

[23] 2019. Privoxy - Home Page. https://www.privoxy.org.

[24] 2019. Secure, Fast & Private Web Browser with Adblocker | Brave Browser. https://brave.com.

[25] 2019. Technical Specifications of Google Pixel 3. https://store.google.com/product/pixel_3_specs.

[26] 2019. TrackStop: Ad, Phishing and Malware Blocker | Perfect Privacy. https://www.perfect-privacy.com/en/features/trackstop.

[27] 2019. Webhiker's Guide to Proxomitron. https://proxomitron.info.

[28] Mika Ayenson, Dietrich Wambach, Ashkan Soltani, Nathaniel Good, and Chris Hoofnagle. 2011. Flash Cookies and Privacy II: now with HTML5 and ETag respawning. *SSRN Electronic Journal* (2011).

[29] Adam Barth, Collin Jackson, and John C Mitchell. 2009. Securing frame communication in browsers. *Communications of the ACM* (2009).

[30] Muhammad Ahmad Bashir, Sajjad Arshad, William Robertson, and Christo Wilson. 2016. Tracing Information Flows Between Ad Exchanges Using Retargeted Ads. In *25th USENIX Security Symposium (USENIX Security 16)*.

[31] Peter Eckersley. 2010. How Unique Is Your Web Browser? *Proceedings on Privacy Enhancing Technologies* (2010), 1–18.

[32] David S. Evans. 2009. The Online Advertising Industry: Economics, Evolution, and Privacy. *Journal of Economic Perspectives* 23, 3 (September 2009), 37–60.

[33] David Gugelmann, Markus Happe, Bernhard Ager, and Vincent Lenders. 2015. An Automated Approach for Complementing Ad BlockersâĂŹ Blacklists. *Proceedings on Privacy Enhancing Technologies* 2015 (02 2015), 282–298.

[34] Saad Sajid Hashmi, Muhammad Ikram, and Mohamed Ali Kaafar. 2019. A Longitudinal Analysis of Online Ad-Blocking Blacklists. In *Proceedings of the 44th IEEE Conference on Local Computer Networks, LCN 2019.*

[35] Muhammad Ikram, Hassan Jameel Asghar, Mohamed Ali Kaafar, Anirban Mahanti, and Balachandar Krishnamurthy. 2017. Towards Seamless Tracking-Free Web: Improved Detection of Trackers via One-class Learning. *Proceedings on Privacy Enhancing Technologies* 2017, 1 (2017), 79–99.

[36] Muhammad Ikram and Mohamed Ali Kaafar. 2017. A first look at mobile ad-blocking apps. In *Network Computing and Applications (NCA), 2017 IEEE 16th International Symposium on.* IEEE, 343–350.

[37] Muhammad Ikram, Rahat Masood, Gareth Tyson, Mohamed Ali Kaafar, Noha Loizon, and Roya Ensafi. 2019. The Chain of Implicit Trust: An Analysis of the Web Third-party Resources Loading. In *Proceedings of the World Wide Web conference (WWW '19)*.

[38] Barabara K. Kaye and Norman J. Medoff. 2001. *Just a Click Away: Advertising on the Internet.* Massachusetts: Allyn and Bacon.

[39] Balachander Krishnamurthy and Craig Wills. 2009. Privacy diffusion on the web: a longitudinal perspective. In *Proceedings of the 18th international conference on World Wide Web (WWW '09).* ACM, 541–550.

[40] Adam Lerner, Anna Kornfeld Simpson, Tadayoshi Kohno, and Franziska Roesner. 2016. Internet jones and the raiders of the lost trackers: An archaeological study of web tracking from 1996 to 2016. In *25th USENIX Conference on Security Symposium (USENIX Security 16).* 997–1013.

[41] Vector Guo Li, Matthew Dunn, Paul Pearce, Damon McCoy, Geoffrey M. Voelker, and Stefan Savage. 2019. Reading the Tea leaves: A Comparative Analysis of Threat Intelligence. In *28th USENIX Security Symposium (USENIX Security 19).*

[42] Sivaramakrishnan Ramanthan, Jelena Mirkovic, and Minlan Yu. 2018. *Blacklists Assemble : Aggregating Blacklists for Accuracy.* Technical Report ISI-TR-730. Information Sciences Institute.

[43] Franziska Roesner, Tadayoshi Kohno, and David Wetherall. 2012. Detecting and defending against third-party tracking on the web. In *9th USENIX Symposium on Networked Systems Design and Implementation (NSDI '12).*

[44] Ashkan Soltani, Shannon Canty, Quentin Mayo, Lauren Thomas, and Chris Jay Hoofnagle. 2009. Flash Cookies and Privacy. *SSRN Electronic Journal* (2009).

[45] Kyle Soska and Nicolas Christin. 2014. Automatically Detecting Vulnerable Websites Before They Turn Malicious.. In *23rd USENIX Conference on Security Symposium (USENIX Security 14).* 625–640.

[46] Marie Vasek and Tyler Moore. 2014. Identifying risk factors for webserver compromise. In *International Conference on Financial Cryptography and Data Security.* Springer, 326–345.

[47] Antoine Vastel, Peter Snyder, and Benjamin Livshits. 2018. Who Filters the Filters: Understanding the Growth, Usefulness and Efficiency of Crowdsourced Ad Blocking. *arXiv preprint arXiv:1810.09160* (2018).

[48] David Y Wang, Stefan Savage, and Geoffrey M Voelker. 2013. Juice: A Longitudinal Study of an SEO Botnet. In *Network and Distributed System Security Symposium (NDSS '13).*

[49] Craig E Wills and Doruk C Uzunoglu. 2016. What ad blockers are (and are not) doing. In *Fourth IEEE Workshop on Hot Topics in Web Systems and Technologies (HotWeb '16).*

[50] Shitong Zhu, Xunchao Hu, Zhiyun Qian, Zubair Shafiq, and Heng Yin. 2018. Measuring and disrupting anti-adblockers using differential execution analysis. In *Network and Distributed System Security Symposium (NDSS '18).*